NVMeTCP10G-IP Demo Instruction

Rev1.1 25-Mar-22

1 Overview

This document describes the instruction to run NVMeTCP10G-IP demo which implements NVMe/TCP Host for accessing one NVMe SSD installed inside the NVMe/TCP Target. This document set up the target system by using Test PC installing Ubuntu 20.04.1 OS. The NVMe SSD on the Target is written or read by the host across 10Gb Ethernet. The test operation to write or read NVMe SSD must be followed the sequence process of NVMe/TCP which is controlled by user through FPGA console.

In the document, Topic 2 shows the example to set up NVMe to run NVMe/TCP target and remove NVMe from NVMe/TCP target on PC Ubuntu OS. Lastly, the operation on FPGA console and the test result are shown in topic 3. More details of each topic are described as follows.

2 PC Setup for NVMe/TCP

2.1 Application Installation

Before running NVMeTCP10G-IP demo, some applications need to be installed on TestPC. These applications require to install once. After that, the installation is not necessary.

```
sudo apt install ethtool
```

1) Ethtool is applied to tune performance network card. Use following command to install ethtools.

```
sudo apt install nvme-cli
```

2) NVMe Command Line Interface (NVMe-CLI) is applied to manage NVMe SSDs in Linux OS. Use following command to install NVMe-CLI.

2.2 Ethernet Interface Setting on TestPC

Before running the demo, it is recommended to set up 10Gb Ethernet network card to get the best performance by following commands. The new terminal must be opened to start Ethernet interface configuration.

1) To list the logical name of 10G Ethernet port on Linux terminal, use following command.

```
sudo lshw -C network
```

Figure 2-1 shows the results when running the command. “enp1s0f0” is the 10Gb Ethernet interface connected to the NVMe/TCP host.

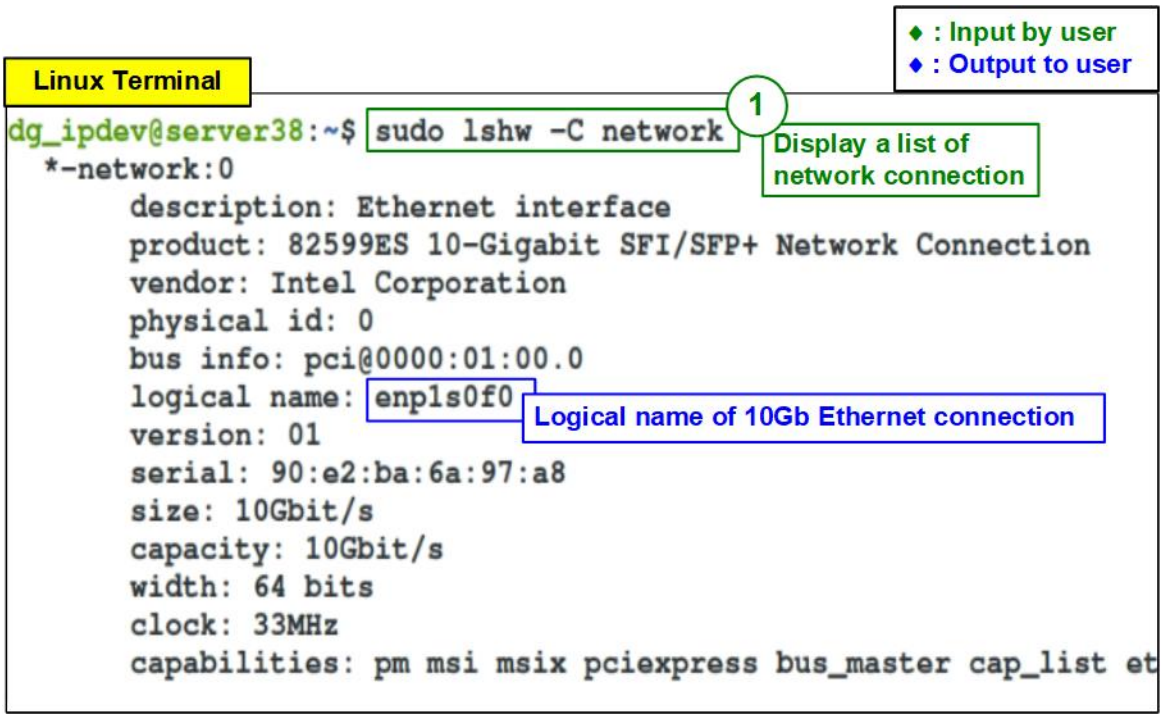

Figure 2-1 Display logical name of 10G Ethernet port

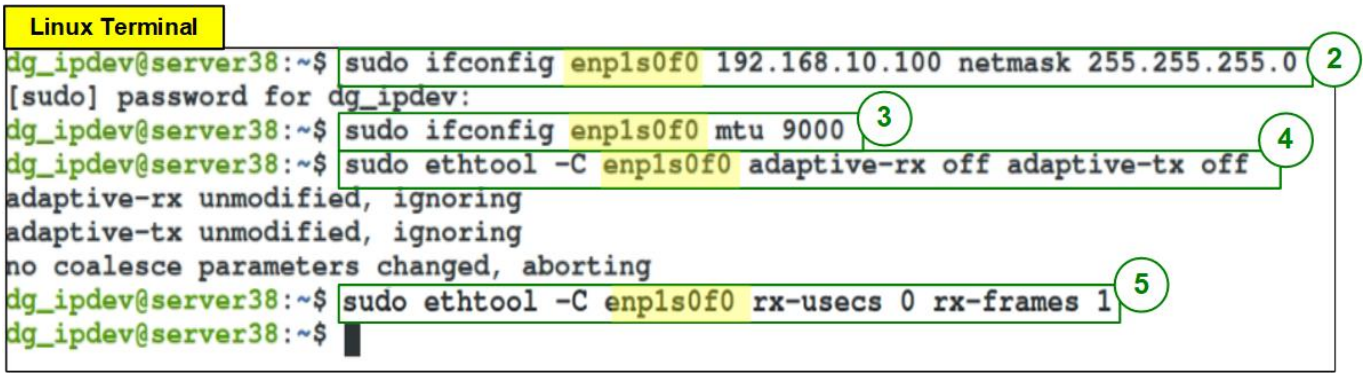

Figure 2-2 IP address and Ethernet interface setting

2) Type “ifconfig <interface> <ipaddr_value> netmask <netmask_value>” to set target IP address and Subnet mask to the desired port of the Ethernet card.
   a) Desired (interface) port of Ethernet card = “enp1s0f0”
   b) Set IP address = 192.168.10.100, Subnet mask = 255.255.255.0

```
sudo ifconfig enp1s0f0 192.168.10.100 netmask 255.255.255.0
```

3) Type “ifconfig <interface > mtu <mtu_value>” to set maximum transfer unit over TCP/IP. It needs to set mtu_value = 9000 to support jumbo frame packet.

```
sudo ifconfig enp1s0f0 mtu 9000
```

4) Turn off Rx-Tx latency improvement algorithm by “sudo ethtool -C <interface> adaptive-rx off adaptive-tx off”.

```
sudo ethtool -C enp1s0f0 adaptive-rx off adaptive-tx off
```

5) Set the highest rate of Rx interrupt by “sudo ethtool -C <interface> rx-usecs 0 rx-frames 1”. This setting executes PC interrupt every time that PC receives a packet.

```
sudo ethtool -C enp1s0f0 rx-usecs 0 rx-frames 1
```

2.3 NVMe/TCP Target setting on TestPC

This topic describes how to configure TestPC to be NVMe/TCP target after finishing 10G Ethernet network setting, as shown in Figure 2-3.

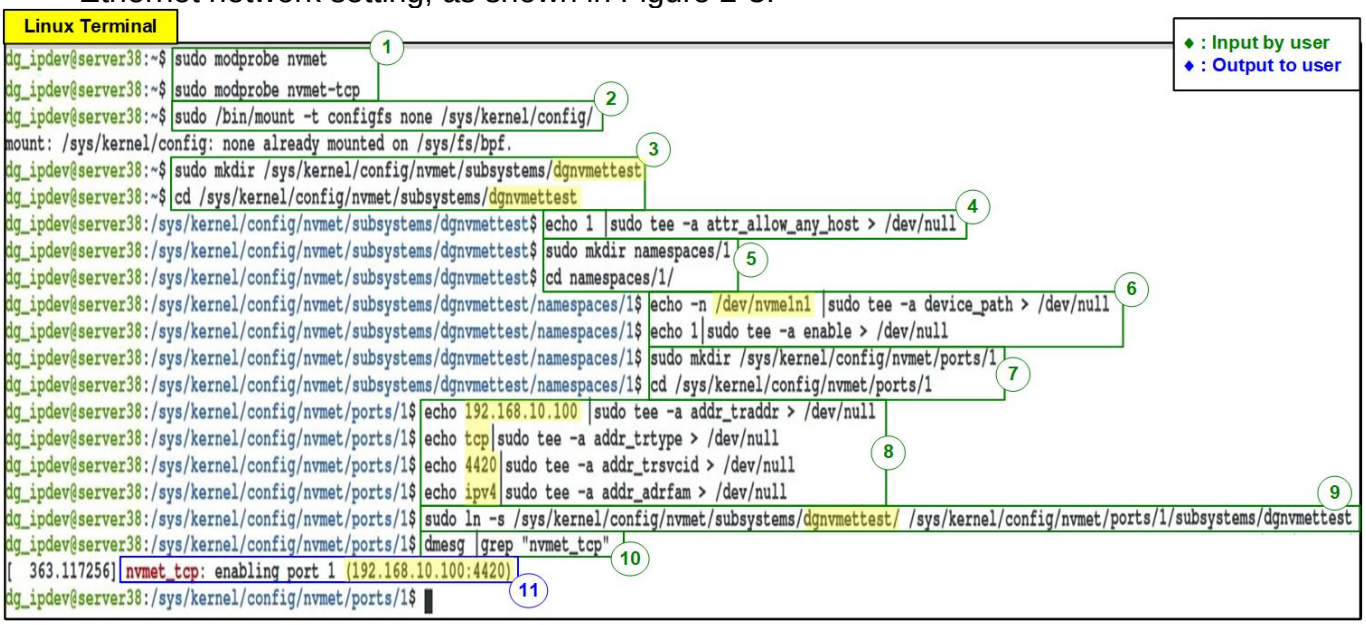

Figure 2-3 Target setting

1) Load the target module by using following command.

```
sudo modprobe nvmet
sudo modprobe nvmet-tcp
```

2) Mount the kernel user configuration filesystem by using following command.

```
sudo /bin/mount -t configfs none /sys/kernel/config/
```

3) Create the target subsystem and define NVMe Qualified Name (NQN). NQN setting in this step must be matched to TrgNQN setting on NVMeTCP10G-IP demo. For example, the name is “dgnvmettest”. After that, change directory to the subsystem directory by following command.
Note: Though TrgNQN on NVMeTCP10G-IP supports up to 256 characters, the demo system allows user to input the name up to 16 characters. Therefore, NQN length set on this step must not be more than 16 characters.

```
sudo mkdir /sys/kernel/config/nvmet/subsystems/dgnvmettest
cd /sys/kernel/config/nvmet/subsystems/dgnvmettest
```

4) Set attribute to allow every host access with the created target subsystem by using following command.

```
echo 1 |sudo tee -a attr_allow_any_host > /dev/null
```

Note: Using "attr_allow_any_host" is the permission for testing only. In the real system, it is recommended to lock the permission by Host NQN.

5) Create a subsystem namespace and change directory to the new directory by using following command.

```
sudo mkdir namespaces/1
cd namespaces/1/
```

6) Set a local NVMe SSD installed in TestPC to the created namespace. After that, enable the namespace by using following command.

```
echo -n /dev/nvme1n1 |sudo tee -a device_path > /dev/null
echo 1|sudo tee -a enable > /dev/null
```

Note: The example of NVMe SSD installed in TestPC is "nvme1n1". The name may be different when running other test environments.

7) Create an NVMe target port to export the created subsystem and change into its directory path as follows.

```
sudo mkdir /sys/kernel/config/nvmet/ports/1
cd /sys/kernel/config/nvmet/ports/1
```

8) Configure Ethernet parameters of the created target port including
   a) IP address = 192.168.10.100
   b) Transport type = "tcp"
   c) Port number = 4420
   d) Address family = "ipv4"

```
echo 192.168.10.100 |sudo tee -a addr_traddr > /dev/null
echo tcp|sudo tee -a addr_trtype > /dev/null
echo 4420|sudo tee -a addr_trsvcid > /dev/null
echo ipv4|sudo tee -a addr_adrfam > /dev/null
```

Note: IP address value corresponds to the IP of Ethernet card port in Topic 2.2 (Ethernet Interface Setting on TestPC).

9) Create a soft link pointed to the target subsystem from the created port by using following command.

```
sudo ln -s /sys/kernel/config/nvmet/subsystems/dgnvmettest/
/sys/kernel/config/nvmet/ports/1/subsystems/dgnvmettest
```

Note: The Target NQN must be corresponded to the previous steps (step 3).

10) Confirm the success of target setting by reading debug message from following command.

```
dmesg |grep "nvmet_tcp"
```

11) If the target is set successfully, the message including target IP address and port number are printed, as shown in Figure 2-4.

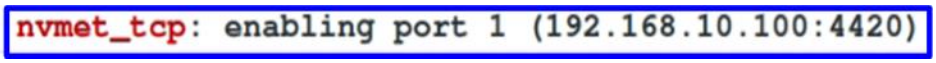

Figure 2-4 NVMe/TCP target setup success message

2.4 NVMe/TCP target removing on TestPC

This topic describes how to remove NVMe from NVMe/TCP target after finishing testing, as shown in Figure 2-5.

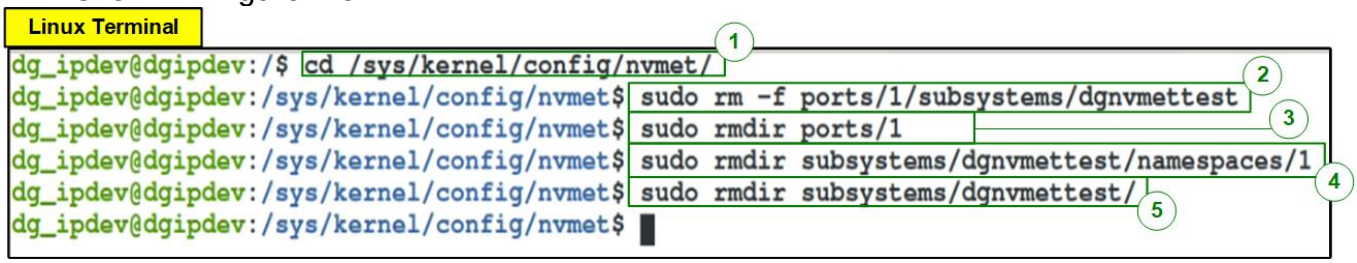

Figure 2-5 Remove NVMe/TCP Target

1) Change directory to nvmet by following command.

```
cd sys/kernel/config/nvmet/
```

2) Remove the target subsystem by following command.

```
sudo rm -f ports/1/subsystems/dgnvmettest
```

3) Remove ports directory by following command.

```
sudo rmdir ports/1
```

4) Remove namespace directory by following command.

```
sudo rmdir subsystems/dgnvmettest/namespace/1
```

5) Remove subsystems directory by following command.

```
sudo rmdir subsystems/dgnvmettest/
```

After finishing removing the target, NVMe SSD is available for CPU access by using general NVMe device driver. User can use “hexdump” to check the updated data inside SSD that is written by the host.

3 Test operation

After finishing PC and FPGA setup, the welcome screen is displayed on FPGA console, as shown in Figure 3-1.

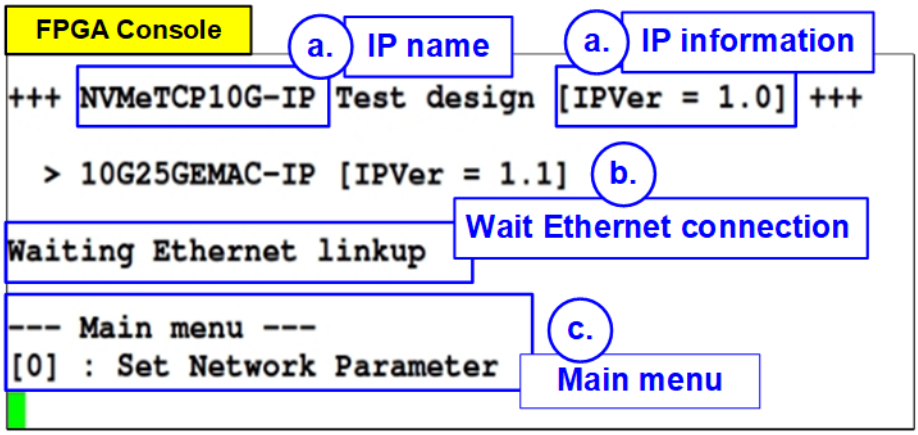

Figure 3-1 Message after system boot-up

a) IP name and IP version number are displayed.
b) Ethernet connection status on FPGA is displayed.
c) After the Ethernet link is established, main menu is displayed. Otherwise, the error message to check the Ethernet cable is displayed, as shown in Figure 3-2.

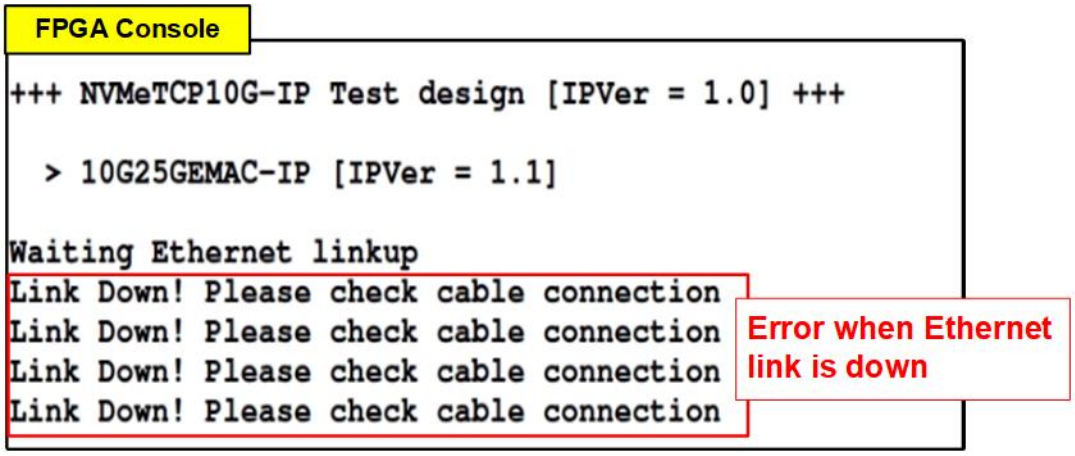

Figure 3-2 Error message when Ethernet connection is down

There are five menus in the test operation. However, only available menu is allowed and displayed on the console in each step for the proper sequence of NVMe/TCP. The first step after system boot-up is menu [0] to set the parameters.

3.1 Set Network Parameter

Select ‘0’ to set the IP parameters.
This menu is used to set IP parameters. After user selects this menu, the current parameters are displayed on the console. User enters ‘x’ to use the same parameters while other keys are entered to change IP parameters. After all parameters are set, the current parameters are displayed again and user will be asked to confirm the parameters. User enters ‘y’ to confirm the parameter. Otherwise, the parameters are denied and user will be asked to set the parameter again.

There are six parameters to set in this menu. Each parameter is verified by CPU. The parameter is updated when the input is valid. If the input is not valid, the parameter does not change. The description of each parameter is shown below.

1) Target NVMe Qualifed Name (NQN): Target NQN indicates which target SSD that the host wants to connect with. The input NQN must not exceed 16 characters, limited by the demo system. Default value is “dgnvmettest”.
2) Host MAC address: 48-bit hex value to be MAC address of the host (NVMeTCP10G-IP). The input is 12 digits of hex value. Add “0x” as a prefix to input as hex value. Default value is 0x000102030405.
3) Host IP address: IP address of the host. The input is a set of four decimal digits is separated by “.”. The valid range of each decimal digit is 0-255. Default value is 192.168.10.1.
4) Host port number: Port number of the host side including Admin port and IO port. Valid range of input is 0-65535. Default value are 40000 and 40001 for Admin port and IO port respectively.
5) Target IP address: IP address of the target. The input is a set of four decimal digits is separated by “.”. The valid range of each decimal digit is 0-255. Default value is 192.168.10.100.

After the confirmation, all parameters are updated to the NVMeTCP10G-IP registers.
“IP parameters are set” is shown on the console and Connect command is available on the main menu as shown in Figure 3-3.

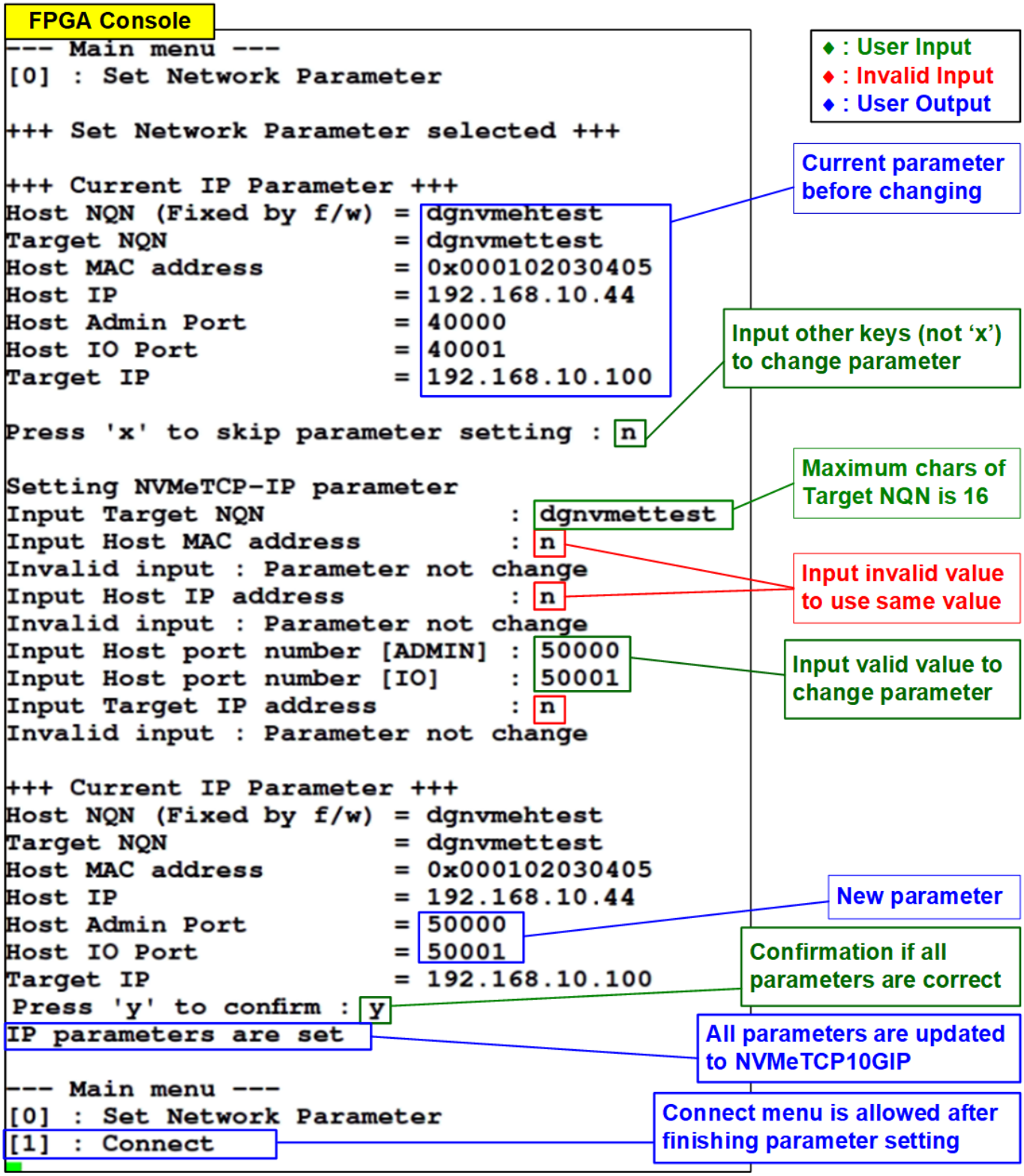

Figure 3-3 Set Network Parameter result

3.2 Connect

Select ‘1’ to connect the host with the target.

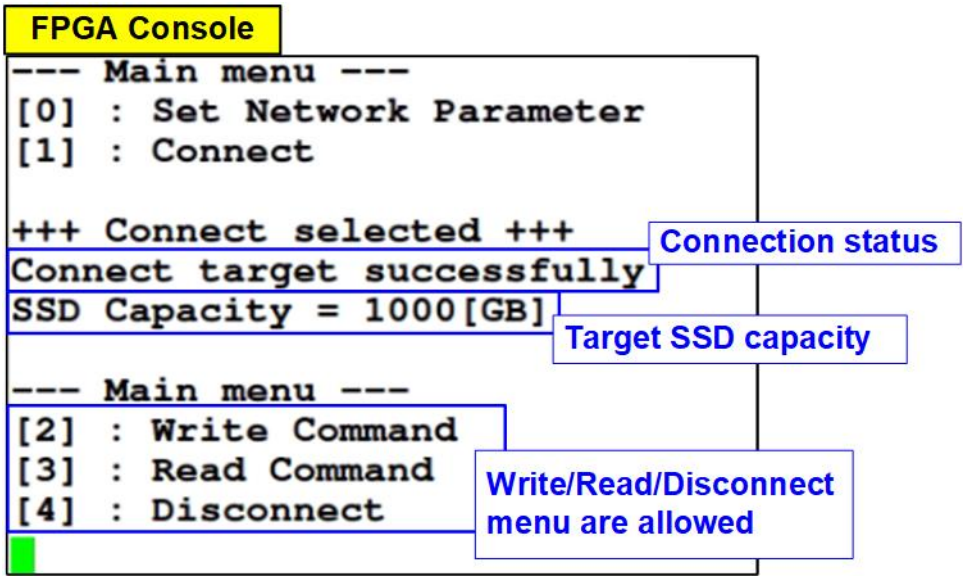

Figure 3-4 Console result when Connect succeeded

This menu is used to initialize the IP by connecting the host (NVMeTCP10G-IP) with desired target indicated via Target NQN. Firstly, TCP/IP connection is established via the network parameters, set from topic 3.1 (Set Network Parameter). After that, NVMe/TCP connection between host and target is established. Once the host successfully connect with the target, “Connect target successfully” and the target NVMe SSD capacity are displayed. Also, Write/Read/Disconnect command are available on the menu, as shown in Figure 3-4.

If the connecting processes employs more than a second, waiting time is displayed in every second. When TCP/IP initialization failed or host failed to connect with the target, error status and error information are displayed. Next, TestPin of the IP is displayed. After error is found, reset system before starting a new test is required as shown in Figure 3-5 and Figure 3-6.

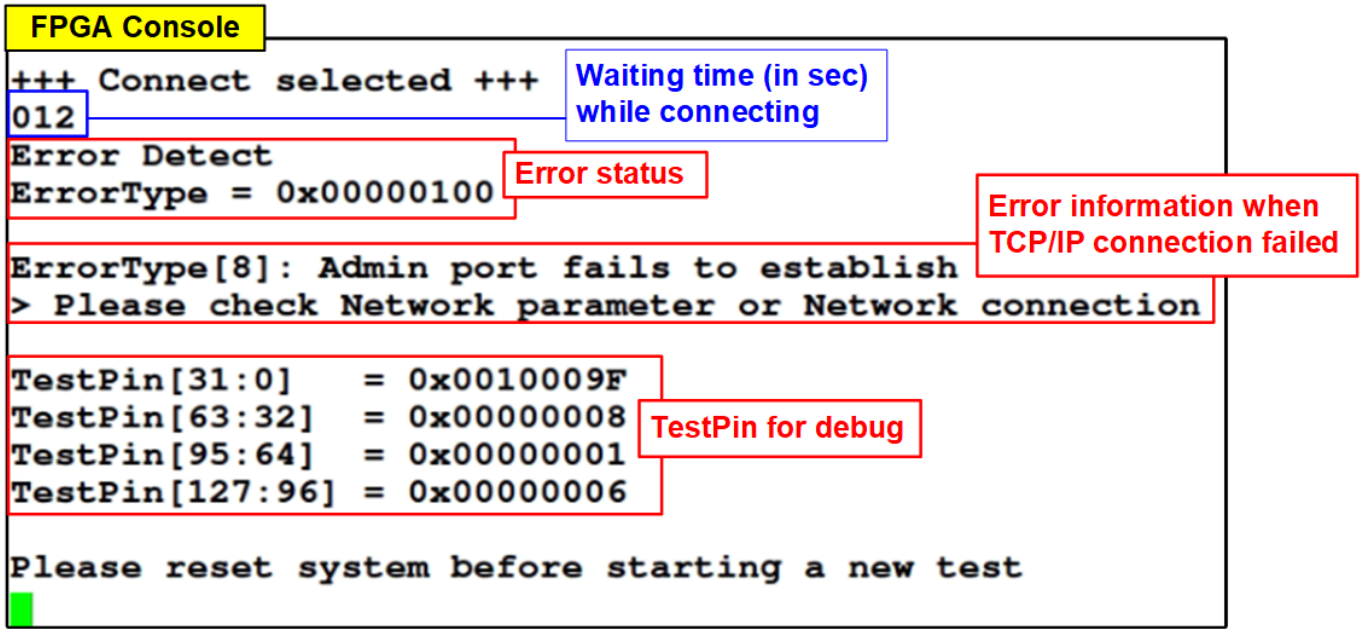

Figure 3-5 Console result when TCP/IP initialization failed

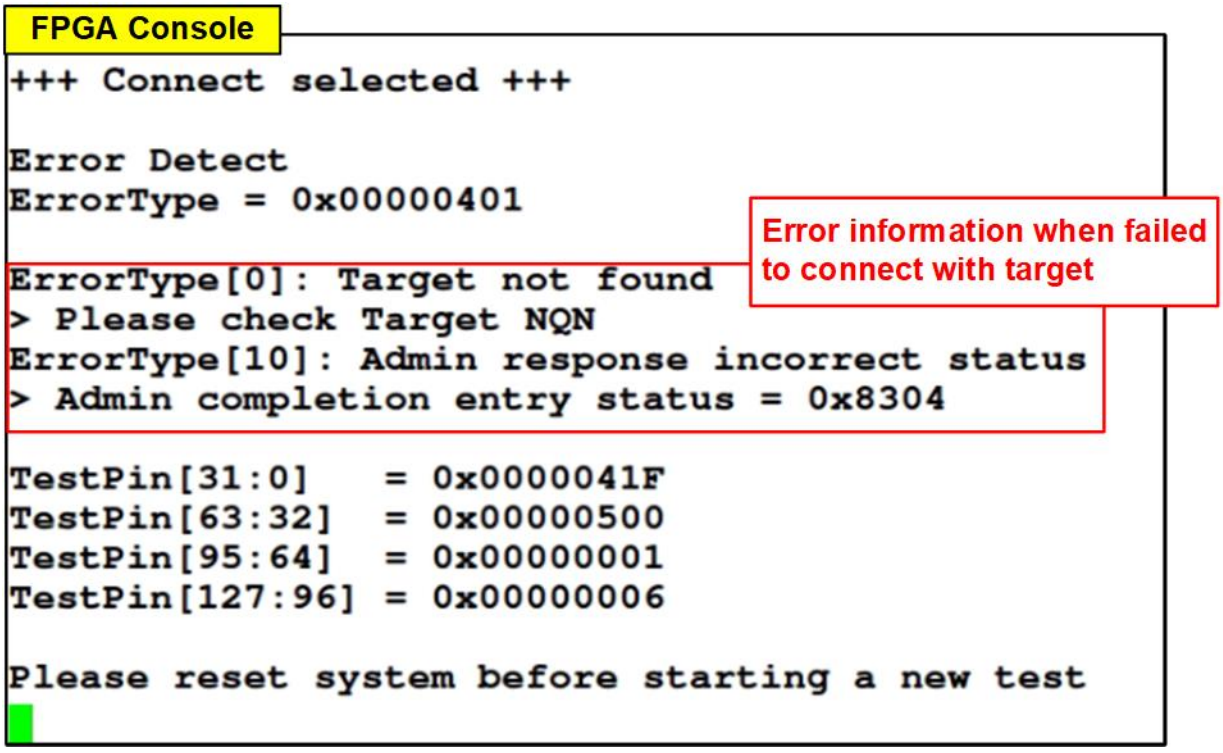

Figure 3-6 Console result when host failed to connect with target

3.3 Write Command

Select '2' to write target NVMe SSD.
This menu is used to test writing operation. The host sends Write command with pattern data across Ethernet to the target for NVMe SSD writing.

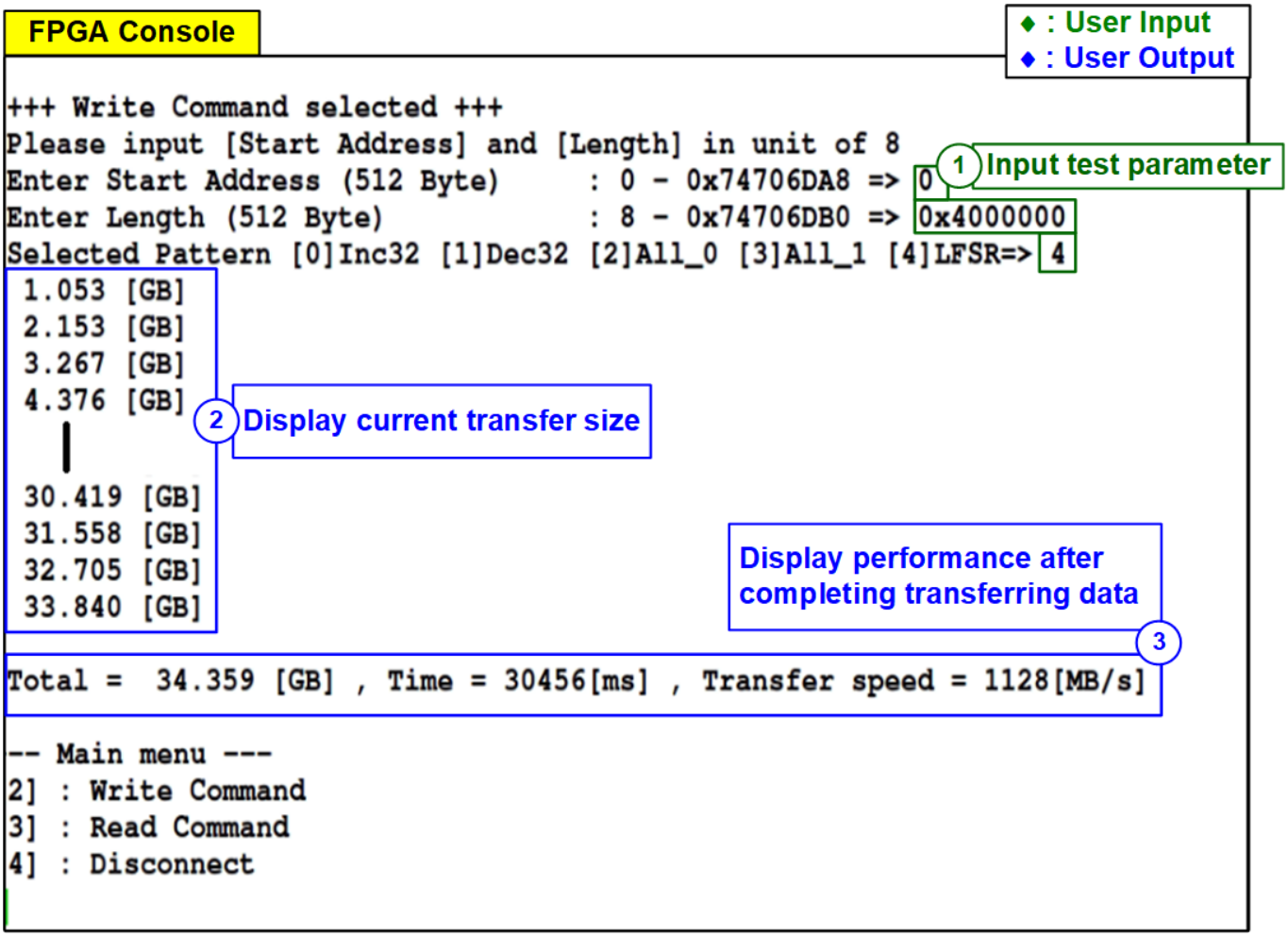

Figure 3-7 Test result when running Write command

User inputs three parameters as follows.

1) Start Address: Input start address to write target NVMe SSD as 512-byte unit. The input is decimal unit when user enters only digit number. User can add "0x" to be prefix for hexadecimal unit. This input must be aligned to 8 because data length of one Write command is fixed to 4096 bytes (8 x 512-byte).
2) Transfer Length: Input total transfer size as 512-byte unit. The input is decimal unit when user enters only digit number. User can add "0x" to be prefix for hexadecimal unit. This input must be aligned to 8 because data length of one Write command is fixed to 4096 bytes.
3) Test pattern: Select test data pattern for writing to target NVMe SSD. There are five patterns, i.e., 32-bit incremental, 32-bit decremental, all 0, all 1, and 32-bit LFSR counter. *Note: Some SSDs shows the best performance when using all 0 pattern.*

When all inputs are valid, the operation begins. During writing data, current amount of transferred data is displayed on the console every second to show that system is still alive. Finally, total size, total time usage, and test speed are displayed on the console as test result.

Test data in NVMe SSD is split into 4096-byte unit. For incremental, decremental, and LFSR pattern, each 4096-byte data has unique 64-bit header consisting of 48-bit address (in 512-byte unit) and 16-bit zero value. The data after 64-bit header is the test pattern which is selected by user. Figure 3-8 shows the example when using 32-bit incremental pattern while Figure 3-9 shows the example when using 32-bit LFSR pattern. The unique header is not included when running all-0 or all-1 pattern.

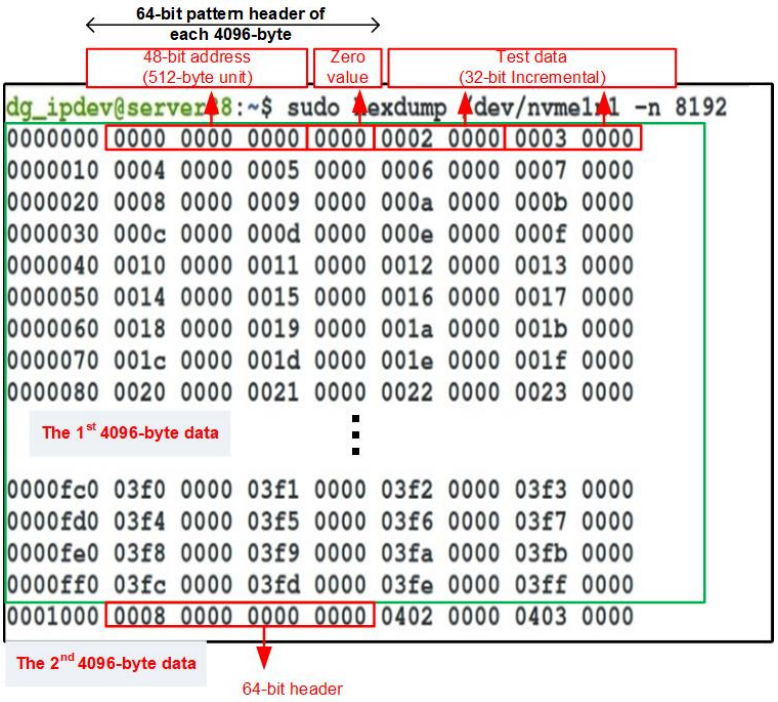

Figure 3-8 Example Test data of the 1st and 2nd 4096-byte by using incremental pattern

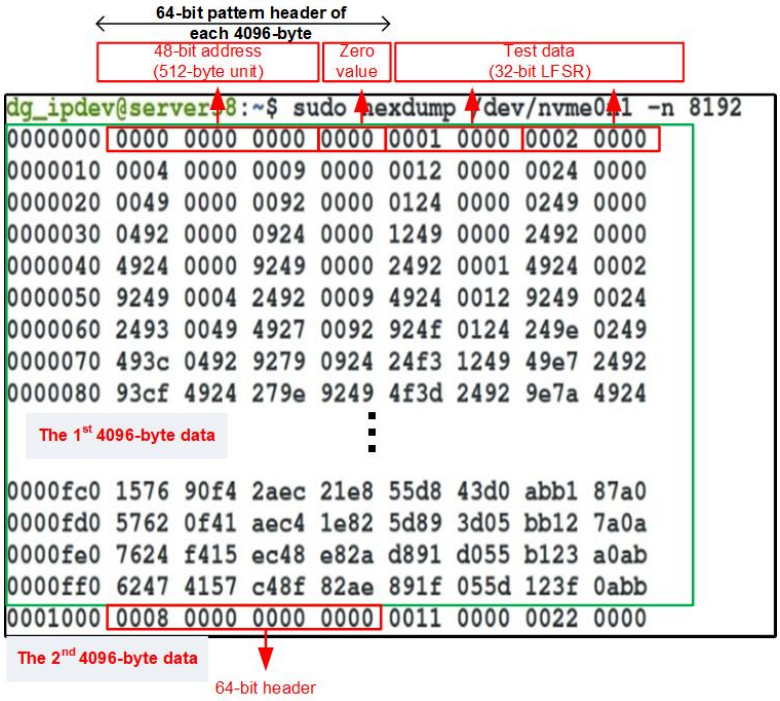

Figure 3-9 Example Test data of the 1st and 2nd 4096-byte by using LFSR pattern

Figure 3-10 show the message when user runs Write or Read command with invalid input. The console displays “Invalid input” and then the operation is cancelled

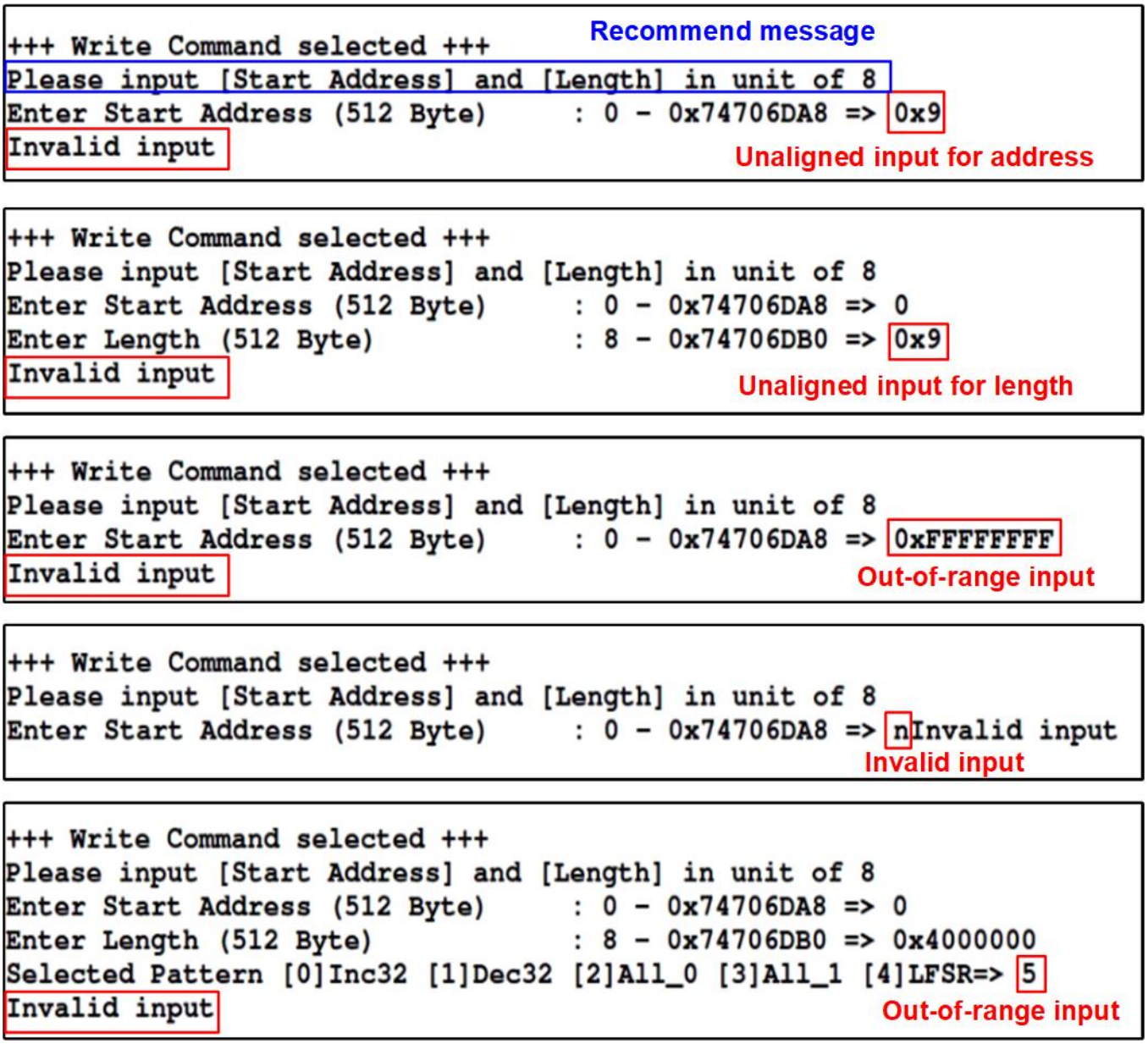

Figure 3-10 Error message when the input is unaligned, out-of-range, or invalid

3.4 Read Command

Select '3' to read target NVMe SSD.
This menu used to test reading operation. The host sends Read command across Ethernet to the target for NVMe SSD reading. Next, the host waits for received data which is read out from target NVMe SSD.

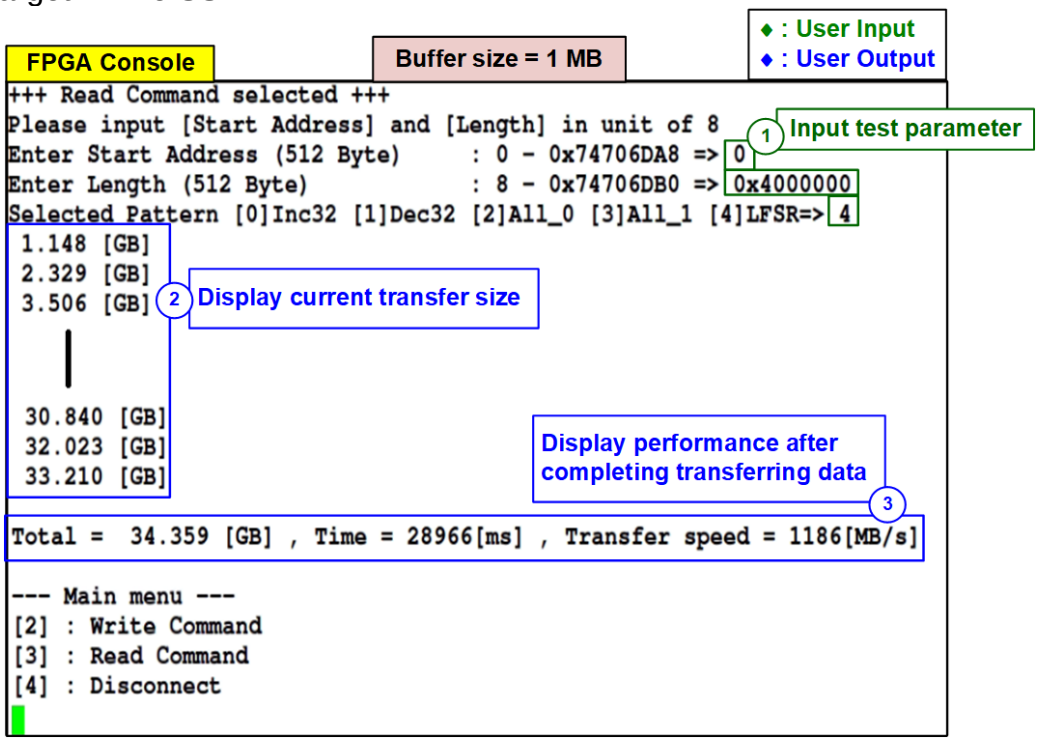

Figure 3-11 Test result when running Read command with 1-MB read buffer size

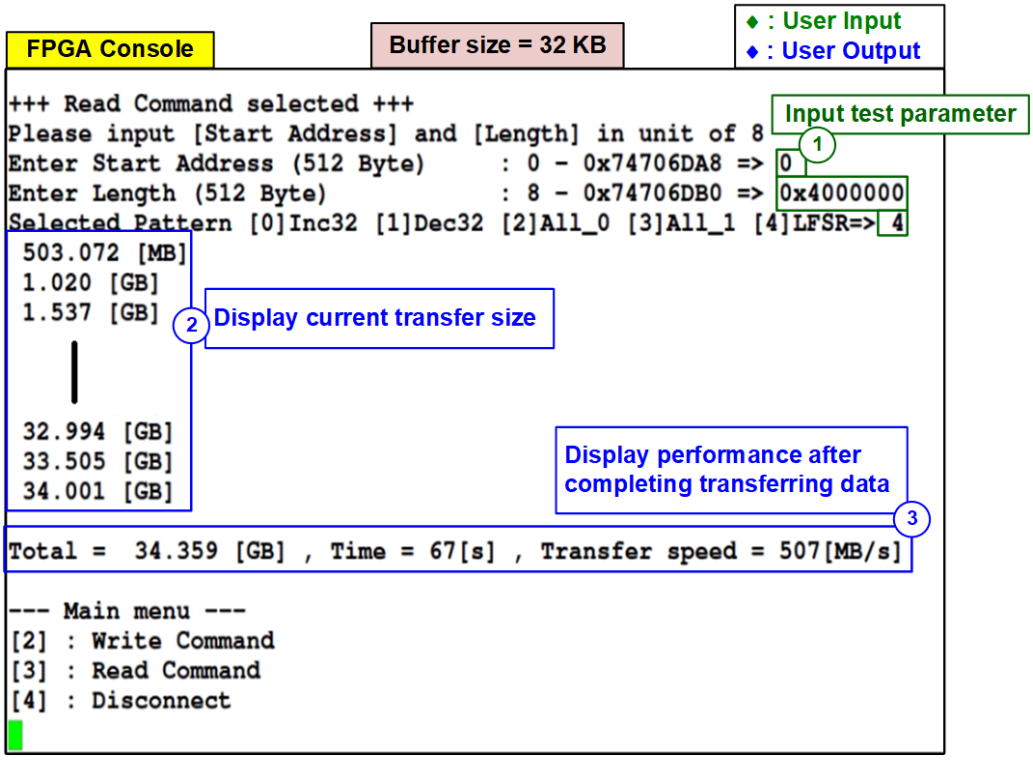

Figure 3-12 Test result when running Read command with 32-KB read buffer size

For Read the target NVMe SSD, user inputs three parameters as follows.

1) Start Address: Input start address to read target NVMe SSD as 512-byte unit. The input is decimal unit when user enters only digit number. User can add “0x” to be prefix for hexadecimal unit. This input must be aligned to 8 because data length of one Read command is fixed to 4096 bytes.
2) Transfer Length: Input total transfer size as 512-byte unit. The input is decimal unit when user enters only digit number. User can add “0x” to be prefix for hexadecimal unit. This input must be aligned to 8 because data length of one Read command is fixed to 4096 bytes.
3) Test pattern: Select test data pattern to verify received data from target NVMe SSD. Test pattern must be matched with the pattern using in Write Command menu. There are five patterns, i.e., 32-bit incremental, 32-bit decremental, all-0, all-1, and 32-bit LFSR counter

Similar to Write command menu, test system reads data from target NVMe SSD when all inputs are valid. During reading data, current amount of transferred data is displayed on the console every second to show that system is still alive. Total size, total time usage, and test speed are displayed after finishing the operation.

Referred from NVMeTCP10G-IP specification, read performance depends on read buffer size as shown in Figure 3-11 and Figure 3-12. Using larger read buffer size may improve the read performance.

Figure 3-13 and Figure 3-14 show error message when data verification is failed without and with cancellation respectively. “Verify fail” is displayed with the information of the 1st failure data, i.e., the error byte address, the expected value, and the read value.

User can press any key(s) to cancel read operation. Otherwise, the operation is still run until finishing Read command. After that, the output performance is displayed on the console.

When cancelling the operation, the read command still runs as the background process and may not finish in a good sequence It is recommended to reset FPGA board. Also, Port connection on target is not closed properly, so it needs to change Admin/IO port number before re-connecting.

FPGA Console

Verification error without cancellation

```
+++ Read Command selected +++
Please input [Start Address] and [Length] in unit of 8
Enter Start Address (512 Byte)      : 0 - 0x74706DA8 => 0
Enter Length (512 Byte)             : 8 - 0x74706DB0 => 0x4000000
Selected Pattern [0]Inc32 [1]Dec32 [2]All_0 [3]All_1 [4]LFSR=> 3

Verify fail
1st Error at Byte Addr = 0x00000000
Expect Data[63:0]      = 0xFFFFFFFF_FFFFFFFF
Read Data[63:0]        = 0x00000000_00000000
Press any key to cancel operation
 1.158 [GB]
 2.369 [GB]
 3.582 [GB]
    |
 31.436 [GB]
 32.637 [GB]
 33.845 [GB]

Total =  34.359 [GB] , Time = 28425[ms] , Transfer speed = 1208[MB/s]
```

Message when verification is failed

Output performance

Figure 3-13 Test result when data verification is failed (without cancellation)

Verification error with cancellation

```
+++ Read Command selected +++
Please input [Start Address] and [Length] in unit of 8
Enter Start Address (512 Byte)      : 0 - 0x74706DA8 => 0
Enter Length (512 Byte)             : 8 - 0x74706DB0 => 0x4000000
Selected Pattern [0]Inc32 [1]Dec32 [2]All_0 [3]All_1 [4]LFSR=> 3

Verify fail
1st Error at Byte Addr = 0x00000000
Expect Data[63:0]      = 0xFFFFFFFF_FFFFFFFF
Read Data[63:0]        = 0x00000000_00000000
Press any key to cancel operation
 1.165 [GB]
 2.378 [GB]
 3.590 [GB]
Operation is cancelled
Please reset system before starting a new test

--- Main menu ---
[2] : Write Command
[3] : Read Command
[4] : Disconnect
```

Message when verification is failed

User enters some keys to cancel the operation

Message when operation is cancelled

Figure 3-14 Test result when data verification is failed (with cancellation)

3.5 Disconnect Command

Select ‘4’ to disconnect host with the target.
This menu is used to terminate the connection between the host (NVMeTCP10G-IP) and the target which is established in topic 3.2 (Connect Command). When user selects Disconnect, the confirmation message is displayed on the console. User enters ‘y’ to continue the operation or enters other keys to cancel the operation.

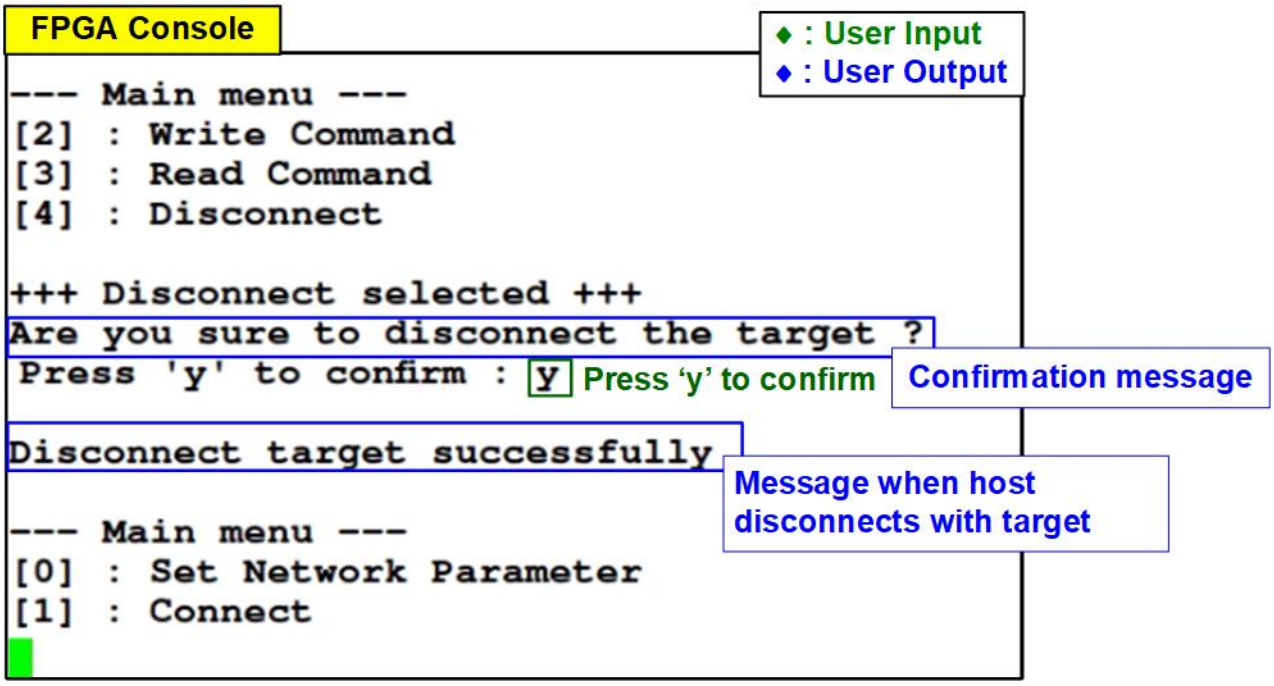

Figure 3-15 Console result when running Disconnect command

Once the host successfully disconnect with the target, “Disconnect target successful” is displayed on the console. At this point, user can change IP parameters by Set Network Parameter or re-connect the host with the previous target by Connect command.

4 Revision History

Revision	Date	Description
1.0	3-Nov-21	Initial version release
1.1	25-Mar-22	Add topic 2.4 NVMe/TCP target removing on TestPC